基于线特征的单目 SLAM 中的迭代数据关联算法 *

魏鑫燏 [1,2]，黄　俊 [2]，杨晓飞 [2]，彭俊杰 [1]

(1. 上海大学 计算机工程与科学学院，上海 200444; 2. 中国科学院上海高等研究院，上海 201210)

摘　要：针对基于线特征的单目 SLAM（同时定位与地图构建）中的数据关联问题，提出了一种基于线段端点 Patch 确认的迭代数据关联算法。算法依据近似共线和端点近似重合两个指标来获取线特征的最近邻关联对，使用基于线段端点 Patch 的定向匹配确认机制来剔除最近邻关联对中的错误项，同时以迭代处理的方式提高数据关联的准确性，综合利用了线特征间的几何约束和图像相似性约束。上述算法在公开数据集上进行了测试，与现有线特征数据关联算法的对比实验结果表明，该算法在满足系统实时性的前提下，在线特征关联对数目和关联准确率上均达到了较好效果。

关键词：同时定位与地图构建；数据关联；线段特征；迭代匹配；特征匹配

中图分类号：TP301.4　　doi: 10.19734/j.issn.1001-3695.2018.05.0508

Iterative data association algorithm for line-based monocular SLAM

Wei Xinyu[1, 2], Huang Jun[2], Yang Xiaofei[2], Peng Junjie[1]

(1. *School of Computer Engineering & Science*, *Shanghai University*, *Shanghai* 200444, *China*; 2. *Shanghai Advanced Research Institute*, *Chinese Academy of Sciences*, *Shanghai* 201210, *China*)

Abstract: Aiming at the data association problem of line-based monocular SLAM, this paper proposed a iterative data association algorithm based on confirmation of endpoint patch. The algorithm obtained closest feature line pairs according to approximate collinearity and approximate coincidence of endpoints, then used directional search of endpoint patches to remove the error pairs. Iterative approach is used to improve accuracy. The algorithm comprehensive used geometric constraint and image similarity of feature line pairs. The experiments are carried out on a public dataset. By comparison with other similar methods, it is proved that the proposed data association approach not only satisfy real-time requirement, but also has a good performance on match amount and matching accuracy.

Key words: simultaneous localization and mapping; data association; line segment feature; iterative matching; feature matching

0　引言

随着增强现实、无人机导航等领域的蓬勃发展，视觉 SLAM 系统在近几年中发展迅速，其中基于特征的视觉 SLAM 系统在准确性和鲁棒性方面的表现尤为出色，涌现出了很多基于特征的视觉 SLAM 系统[1,2]，这些视觉 SLAM 系统中最常使用的特征是点特征，但基于点特征的 SLAM 系统在纹理稀疏的场景中（如人造环境中的室内、走廊等）存在特征数量较少的问题；同时由于在数学意义上点的维度为零，基于三维点特征的地图无法表现更多层次的结构化信息，在增强现实等应用中常常需要在此基础上再次提取平面、直线等特征。部分学者尝试在视觉 SLAM 系统中使用其他特征，这些系统中又以基于线特征的系统为主流[3~6]，其原因在于一方面基于线特征的三维地图能够提供更多环境结构信息；另一方面在视觉 SLAM 应用领域中纹理稀疏的场景常常具有较为丰富的结构化特征（走廊、空旷的房间），而这些场景中可以提取的线特征数目较多。

相比于点特征，线特征 SLAM 的发展速度相对缓慢，而且目前已有的一些单目线特征 SLAM 系统均存在一定问题：这些 SLAM 一部分对应用环境有特殊要求[4,7,8]；一部分仅能处理线段极其稀疏的场景[3,5, 9]。以上这些问题出现的一个重要原因在于目前单目线特征 SLAM 中数据关联部分存在一定困难。

数据关联是 SLAM 问题中的重要难点之一，对 SLAM 系统的鲁棒性和稳定性影响极大，数据关联是指在 SLAM 系统中的地图数据和当前传感器的输入数据之间建立关联的过程。在视觉 SLAM 系统中，点特征的数据关联问题已经研究地较为透彻，但由于线特征本身存在的端点不稳定、过度分割等问题，线特征的数据关联还有待进一步研究。

目前单目线特征 SLAM 中并没有一个通用的数据关联方

收稿日期：2018-05-21；**修回日期：**2018-08-22　　**基金项目：**国家自然科学基金资助项目（61201446）

作者简介：魏鑫燏（1992-），男，湖北人，硕士研究生，主要研究方向为视觉 SLAM（weixy@sari.ac.cn）；黄俊（1980-），男，浙江人，副研究员，主要研究方向为计算机视觉、模式识别；杨晓飞（1975-），男，四川人，高级工程师，主要研究方向为物联网工程；彭俊杰（1977-），男，湖北人，副教授，主要研究方向为云计算、光学计算机.

法，但已有的算法大致可以分为如下三类：第一类利用当前相机预测位姿将地图中的三维线段投影到二维图像平面上，使用最近邻匹配的方法得到三维线段与二维线段的匹配对[4]；第二类使用二维线段端点或其他采样点周围的图像块（Patch）[8,9]和线段描述子[10]来获取关联对，将三维线段已关联的二维线段的线段描述子或采样点 Patch 与待关联的二维线段进行比较，得到关联对，其中线段描述子以 MSLD[11]和 LBD[12]为主；第三类方法结合了前两种数据关联方法[3,5]，使用最近邻方法得到候选二维线段集合，然后使用基于线段端点 Patch 或线段描述子的方式得到最终匹配结果。其中第一类方法的速度最快，但在线段密集或预测位姿误差较大的情况下匹配错误率过高，而且如何判断最近邻线段也是一个没有很好解决的问题；第二类方法准确度较高，但其计算量较大，同时由于目前已有的线特征提取算法得到的线段端点可重复性较差的问题，仅使用图像特征难以得到足够的匹配对；第三类方法虽然改善了以上单一方法存在的问题，但已有的实现方式中计算量仍然较大，当预测位姿误差较大时，最后得到的正确匹配对数目仍然较少。

本文针对单目线特征 SLAM 中地图内三维线特征与图像中二维线段的关联问题展开研究，提出了一种基于线段端点 Patch 确认的迭代线特征关联算法。该算法融合了最近邻匹配算法和线段端点 Patch 匹配算法，在改进最近邻匹配中的最近邻线段定义、Patch 匹配中 Patch 搜索策略的基础上，通过 Patch 匹配来减小最近邻匹配方法错误率较高的缺陷，利用迭代优化的框架处理关联算法中输入数据噪声较高的问题。

1　问题定义

本文讨论的数据关联问题的应用场景是基于图优化框架的单目线特征 SLAM 系统，本文将单目线特征 SLAM 系统中的数据关联问题定义为：给定一个三维线特征集合 $L=\{L_1,L_2,\cdots,L_a\}$ 和当前帧 I_C 内的二维线段集合 $\hat{l}=\{l_1,l_2,\cdots,l_b\}$，在 L 与 $\hat{l}$ 两个集合中寻找正确匹配对的问题。

在 SLAM 系统中，数据关联模块的主要功能是为相机位姿和地图三维特征的优化过程提供输入数据，将数据关联算法的结果表示为 $G=\{g_1,g_1,\cdots,g_n\}$，其中 $g_i(i=1,2,\cdots,n)$ 表示某个特定的关联对 (L_i,l_j)，关联对中三维线特征与二维线段的关联距离表示为 $d(g_i,T_c,K)$，其中 T_c 为当前帧位姿，K 为相机内参矩阵，将关键帧位姿集合表示为 $T=\{T_1,T_2,\cdots,T_m\}$，SLAM 系统中以下两个优化过程会中使用到数据关联结果：

$$T_c=\arg\min_{T_c}\sum_{g_i\in G}d(g_i,T_c,K) \tag{1}$$

$$(L,T)=\arg\min_{(L,T)}\sum_{g_i\in G}\sum_{T_j\in T}d(g_i,T_j,K) \tag{2}$$

其中式（1）用于优化当前相机位姿，式（2）用于优化地图中的关键帧位姿和三维线特征，以上两个优化过程的原理基本相同，不失一般性，下文以优化当前相机位姿的过程为例阐述数据关联算法。

2　基于 Patch 确认的迭代线特征关联算法

基于 Patch 确认的迭代线特征关联算法首先基于最近邻线段定义得到候选匹配对，然后利用固定方向的图像块搜索策略剔除错误匹配，之后利用已得到的匹配对优化当前相机位姿，最后将上述过程进行迭代优化。下面对算法的各个步骤进行详细阐述。

2.1　基于最近邻匹配的初步数据关联

初步关联结果的获取可分为两个步骤：缩小候选关联集合、确定最近邻匹配对象。详细介绍如下：

a）缩小候选关联集合。最近邻匹配中首先将三维线特征投影到当前相机平面得到二维投影线段，然后根据投影线段计算出图像中的候选关联线段集合。为加快匹配速度，本文利用线段的角度和距离属性来缩小候选关联集合。三维直线投影过程采用文献[13]中基于 Plücker 坐标的投影方法，在获取三维直线的二维投影线段之后，为提高投影线段对应的候选关联线段的搜索速度，将图像中的二维线段被划分为多个集合。本文使用线段角度和线段所在直线到图像中心点的距离两个属性来确定候选匹配集，其中线段角度指的是二维线段所在直线与图像坐标系 x 轴正方向的夹角。二维线段按线段角度（0-90 度）被分为 n 个集合 $l_i^{\mathrm{A}}(i=1,2,\cdots,n)$，每个集合中的最大角度差值为 $90/n$ 度；按线段所在直线到图像中心点的距离二维线段被分为 m 个集合 $l_i^{\mathrm{D}}(i=1,2,\cdots,m)$，在图像长宽分别为 640 像素和 480 像素的数据集中，每个集合中的线段到图像中心的距离最大差值为 $400/m$ 像素。对于待匹配三维线段的二维投影线段 l_L，首先得到其角度值 β 和到图像中心点的距离 d，由 β、d 及对应属性差值的阈值 $\beta_{\max}$、$d_{\max}$ 分别确定候选集合 $l_{\mathrm{t}}^{\mathrm{A}}$ 和 $l_{\mathrm{t}}^{\mathrm{D}}$，最终的候选二维线段集合为 $l_{\mathrm{t}}=l_{\mathrm{t}}^{\mathrm{A}}\cap l_{\mathrm{t}}^{\mathrm{D}}$。需要说明的是虽然同时使用了线段的两个属性进行分类，其分类算法的时间复杂度仍为常数阶，如果仅使用单一属性，在同向线段较多和线段分布不均匀的场景中候选匹配集合会十分庞大，而这将大大增加后续步骤的计算时间。

b）确定最近邻匹配对象。在得到候选二维线段集合之后，需要在此集合中确定最近邻匹配线段，这一过程中的重点在于最近邻线段的定义。与点特征不同，定义最近邻线特征是一个较为复杂的问题[14]，由于通常使用线段端点到另一线段所在直线的距离来表示线段间距离，而这一数据是多维的，无法直观判断出最近邻线段，需要根据线段自身的属性对最近邻线段进行计算。将三维线段的二维投影线段表示为 l_L，图像中提取出的二维线段为 l_1、l_2，定义二维线段间距离 $d(l_L,l)$ 如下：

$$d_{\mathrm{P}_1 l_L}(l_L,l)=a^{\mathrm{T}}\cdot l_L\Big/\sqrt{(l_L^1)^2+(l_L^2)^2} \tag{3}$$

$$d_1(l_L,l)=0.5\left(d_{\mathrm{P}_1 l_L}(l_L,l)+d_{\mathrm{P}_2 l_L}(l_L,l)\right) \tag{4}$$

$$d_2(l_L,l)=0.5\left(d_{P_1}(l_L,l)+d_{P_2}(l_L,l)\right) \tag{5}$$

$$d(l_L,l)=\left(d_1(l_L,l)+d_2(l_L,l)\right)\frac{l_A^*}{l^*} \tag{6}$$

其中：$d_{P_1 l_L}(l_L,l)$ 和 $d_{P_2 l_L}(l_L,l)$ 表示线段 l 的两个端点到投影线段 l_L 所在直线的距离，采用与文献[8]中相同的计算方法，以 $d_{P_1 l_L}(l_L,l)$ 为例，式（4）中 a 为线段 l 的一个端点（用齐次坐标表示），l_L 所在直线表示为 $(l_L^1,l_L^2,l_L^3)^{\mathrm{T}}$；$d_{P_1}(l_L,l)$ 和 $d_{P_2}(l_L,l)$ 表示线段 l 和线段 l_L 间对应端点的距离，l_A^* 表示当前图像平面内全部二维线段的平均长度，l^* 表示线段 l 的长度，l_A^*/l^* 参数可以使在位姿优化过程中贡献更大的长线段更容易被匹配。本文将线段的临近属性定义为两类：一类是共线属性，用端点到直线的距离，即 $d_1(l_L,l)$ 量化表示这一属性；一类是位置重合属性，表示为对应端点间的距离 $d_2(l_L,l)$。典型的线段距离定义一般仅以 $d_1(l_L,l)$ 的形式出现，在线段密集的场景中传统定义方法无法分辨共线或近似共线的线段，而仅使用端点间距离又无法鲁棒地处理线段端点不稳定的情况。

在得到候选二维线段集合之后，对集合中的每一条二维线段 l 分别计算其与三维线特征投影线段的距离 $d(l_L,l)$，选取距离最小的二维线段，如果其最小距离小于阈值 $d_{\max}^L$，则将此二维线段作为三维线特征 L 的关联特征。在实际操作中，本文采用的参数为 $n=18$、$m=20$、$\beta_{\max}=6$、$d_{\max}=40$、$d_{\max}^L=90$。

2.2　基于线段端点 Patch 的错误关联剔除

由于 SLAM 系统中三维线段和当前帧预测位姿的误差，最近邻匹配算法得到的匹配对中不可避免地存在错误关联，这些错误关联将严重影响当前帧位姿计算结果的精度，本文采用基于线段端点图像 Patch 的错误匹配剔除策略来处理这些错误关联。

Patch 为二维点附近的一个正方形图像块，使用线段上端点或中点附近的 Patch 来进行线段匹配是一个较为常用的做法[8,15]，这类方法可以简述为如下步骤（以线段端点为例）：

a)获取三维线特征对应的 Patch，此处一般使用已经关联到该三维线特征的二维线段端点附近的图像 Patch；

b)将三维线特征投影到当前图像平面，在候选二维线段集合中进行搜索，计算每一条候选二维线段的端点 Patch 与三维线特征对应 Patch 之间的 ZMSSD（零均值像素灰度差平方和）值，将 ZMSSD 值最小且小于阈值的二维线段关联到该三维线段。

基于端点附近图像 Patch 的匹配方案利用了线段端点处的图像相似性约束，但由于单目线特征 SLAM 中相机预测位姿误差相对较大，而且线段端点可重复性较差，此方案能够得到的关联对过少，在线段较为密集的场景中，此方案的计算量会明显增加。

本文将基于线段端点 Patch 的匹配方法用于最近邻匹配结果的错误剔除上。具体步骤为：

a)获取三维线特征对应 Patch；

b)在三维线特征关联的二维线段端点的“松弛范围”（“松弛范围”的定义请见下文）内沿与“松弛范围”共线的方向进行 Patch 搜索，如果搜索到的图像 Patch 与原三维线特征对应 Patch 间的 ZMSSD 值小于阈值，则确认此关联正确，否则剔除掉此错误关联。

在获取三维线特征对应 Patch 的过程中，首先需要选择三维线特征对应的关键帧 F_L^b，将与当前帧中三维线特征 L 的观测角最相近的关键帧作为最佳观测关键帧 F_L^b。其中观测角定义为当前帧相机主轴与过相机光心和三维直线的平面的夹角，如图 1。其中 L_z 为相机主轴，即过光心 C 垂直于图像平面的直线，π_L 为过光心 C 和三维直线 L 的平面，观测角为 L_z 与 π_L 的夹角 α。取得关键帧之后从其中 L 对应的二维线段 l_L 的端点处取得两个 8*8 的图像块，后续处理过程则与 PTAM[1]中的 Patch 提取过程完全一致，其实现细节可参考文献[1]。

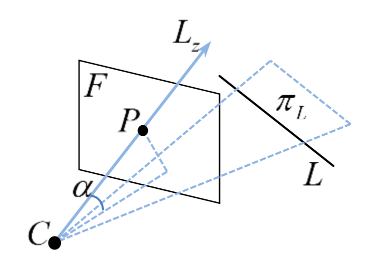


图 1　三维直线在当前帧中的观测角

Fig.1　Viewing angle of 3D lines in current frame

Patch 搜索阶段中计算候选二维线段端点的图像 Patch 与投影线段端点图像 Patch 间的 ZMSSD 值，当 ZMSSD 值小于阈值时，关联被判定为正确。但由于线段端点不稳定，这种方法可能会导致大量正确关联被剔除。为解决这一问题，本文定义了线段端点的“松弛范围”，“松弛范围”为以原线段端点为中心、与原线段共线的一段长度为 S 个像素的线段，如图 2，其中 F_2 帧中 a_2b_2 为原线段，端点 a_2 的“松弛范围”为线段 a_3b_4，其长度 S 定义如下：

$$S=\begin{cases}10 & l_L^*\geq 20\\ l_L^*/2 & l_L^*<20\end{cases} \tag{7}$$

其中：l_L^* 为原二维线段的长度，S 的最大值为 10 个像素，当 l_L^* 过小时，S 的值随 l_L^* 而变换。“松弛范围”使得错误剔除算法更为鲁棒，当线段端点的误差未超出“松弛范围”时，正确的关联对不会被错误剔除。

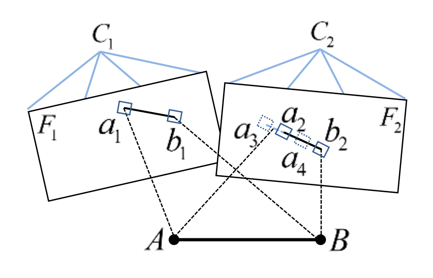


图 2　松弛范围内的端点 Patch 确认

Fig.2　Determination of endpoint-patch in relaxation range

2.3　迭代关联算法

迭代关联算法中使用上述最近邻匹配方法得到候选匹配对，然后采用基于线段端点 Patch 的错误关联剔除算法去除错误关联对，利用得到的关联对优化当前帧位姿，最后将以上过程迭代直至优化过程中的匹配错误率小于阈值。

在位姿优化过程中，目标函数表述为式（1），为避免线段端点不稳定对优化过程带来的不利影响，目标函数中的距离函数 $d(g_i,T_c,K)$ 采用式（3）中描述的线段端点到直线的距离，本文使用 g2o[16]中的 LM 算法来实现对相机位姿的优化。迭代运算结束的判定标准为相机位姿优化过程中的内点比例大于最近关键帧中内点比例的 70%，其中内点定义为位姿优化后距离小于阈值的关联对，同时考虑到 SLAM 系统的实时性，最大迭代次数定义为 3 次。

基于 Patch 确认的迭代线特征关联算法如下所示。

算法 1 基于 Patch 确认的迭代线特征关联算法

输入：二维线段集合 $l=(l_1,l_2,\ldots,l_n)$，三维线段集合 $L=(L_1,L_2,\ldots,L_m)$，当前相机位姿 P

输出：三维线段与二维线段匹配对 G

```
function IterativeLineAssociation(l,L,P)
    for i = 0 → 3 do
        G ← LineAssociation(l,L,P)
        P ← OptimizePose(P,G)//按照公式 1 优化相机位姿
        if Inliers(P,G) then      //根据内点数目判断是否结束循环
            break
    end for
    return G
end function

function LineAssociation(l,L,P)
    for i = 0 → m do
        pt ← GetPatch(L[i])          //获取三维线段端点 Patch
        lp ← Project3DLine(l[i],P)   //将三维线段投影到图像平面
        ls ← Get2DLineSet(lp,l)      //获取缩小后的候选线段集合
        fNear ← false
        for j = 0 → len(ls) do
            d ← Dis(ls[j],lp)          //按照公式 6 计算线段间距离
            if d < Dth then            //距离满足阈值
                fNear ← true
                lb ←ls[j]
                Dth ← d
            end if
        end for
        if fNear then                //找到最近邻线段
        fPatch ← MatchPatch(pt,lb) //按照 2.2 节进行 Patch 确认
        if fPatch then
            result ← (L[i],lb)
            end if
        end for
    return result
end function
```

3　实验及结果分析

3.1　实验设计

为验证本文所提出的迭代线特征关联方法的准确性和高效性，与同类算法进行了对比，并在公开数据集上进行了相关实验。

本文采用如图 3 所示的实验方案。实验所用 SLAM 系统采用与 ORB-SLAM[2]相同的系统框架，将其中的点特征操作替换为了对应的线特征操作，并去除了闭环检测模块，使用速度较快的 EDLine[17]算法提取图像中的二维线段。同时对原系统中跟踪线程的局部地图跟踪模块做了改动，并在此模块后增加了关联误差评价模块。局部地图跟踪模块中同时使用了三种线特征关联算法，将待处理的数据同时输入到三种算法中，算法分别如下：

算法 1 为本文提出的数据关联算法；

算法 2 为基于最近邻线段的迭代匹配算法，即在本文算法的基础上去除 Patch 匹配确认的步骤，在文献[5]中采用；

算法 3 为基于 LBD 描述子的关联算法，在文献[10]中采用。

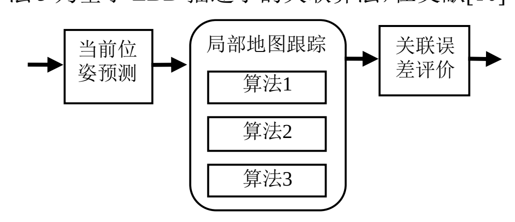


图 3　数据关联算法对比实验设计

Fig.3　Design of contrastive experiment between different data association algorithms

由于 SLAM 系统内三维地图是实时建立和更新的，所以数据关联算法的输入数据中不可避免地存在误差项，关联误差评价模块很难把这部分误差项完全剔除。为了准确评价 SLAM 应用中数据关联算法的准确性和效率，本文实验方案中三种算法在每一帧的输入数据完全相同，关联误差评价模块分别对三种线特征关联算法的输出数据使用相同的方法进行评价。本文采用文献[18]中的参数作为评价指标，包括匹配线段长度比率 M、平均投影误差 G 和最小角度区内匹配线段长度之和 L。具体计算方法如下：

$$M = \frac{\sum_{i,j} l_{ij}}{\min(\sum_i \mathrm{I}^i, \sum_j \hat{\mathrm{I}}^j)} \tag{8}$$

其中: l_{ij} 表示关联算法获得的关联对中二维线段的长度，I 表示图像中提取的二维线段的长度，$\hat{\mathrm{I}}$ 表示可以投影到当前帧的三维线段在图像内的二维投影线段的长度。匹配线段长度比率 M 用于描述算法是否能够获得足够多的匹配对。

$$G = 0.5\frac{\sum_{i,j}\left(d_{P_1 l_L}\left(l_L^i, l^j\right) + d_{P_2 l_L}\left(l_L^i, l^j\right)\right) l_{ij}}{\sum_{ij} l_{ij}} \tag{9}$$

平均投影误差 G 表述了线特征关联算法所获取的关联对的平均 L 投影误差。

$$L = l_1 + l_2 + l_3 \tag{10}$$

其中: l_1、l_2、l_3 分别表示关联对中不同角度区间内二维线段的长度之和。本文将二维线段的角度限定为 0~180 度之间，按照文献中类似的处理方法将角度分为 4 个区间：表示水平方向线段的区间，角度范围包括（0°，22.5°）和（157.5°,180°）；表示垂直方向线段的区间，角度范围为（67.5°,112.5°）；其余方向则分为两个区间（22.5°,67.5°）和（112.5°,157.5°），每个区间内的已关联的线段长度之和表示为 $l_i (i = 1,2,3,4)$，去除其中最长的 l_4，将剩下的 3 项相加得到指标 L，用于表示关联算法在处理不同方向线段时的性能。

文献[18]中为以上三个参数分别设置了相应阈值，当有任意一项指标无法满足阈值要求时判断此帧数据关联失败，本文采用相同的方法来计算关联错误的帧率。由于文献[18]是基于双目摄像头的系统，系统中可以直接得到三维线段的深度信息，而本文的单目系统中无法直接获取深度信息，三维线段的误差相对较大，这使得数据关联算法的输入数据中存在较大误差，因此本文中 G 和 M 两项评价指标采用了更宽松的阈值设置，具体为 $L_{min} = 100px$、$G_{max} = 4.5$ 和 $M_{min} = 0.1$。

除以上指标之外，本文还对不同数据关联算法的在每帧的运算时间均值和得到的关联对数目均值进行了统计，同时为了进一步对算法进行分析，实验中记录了参数 G 相对于每一帧的平均值。

3.2 实验结果及分析

实验平台中 CPU 型号为 Intel i7-3700，3.4GHz，4G 内存。实验数据采用 TUM RGBD 数据集[19]，本文选取了数据集中的四个场景作为数据关联算法的实验数据，见图 4，其中从上到下，从左至右分别为 fr2_xyz、fr3_long_office、fr3_str_tex_far、fr2_rpy 场景中的典型图像帧，按照此顺序将四个场景分别命名为场景 1-4。



图 4　TUM RGB-D 数据集中的典型图像帧

Fig.4　Typical frame samples in TUM RGB-D dataset

实验中，分别将数据集中的图像输入到单目线特征 SLAM 系统中，依照 3.1 节中介绍的实验设计方案，在局部地图跟踪模块中依次运行三种线特征关联算法，其中输入数据为当前局部地图中的三维线段集合以及当前帧内二维线段集合，输出数据为三维线特征与二维线段的匹配对，记录每种数据关联算法的四项指标。

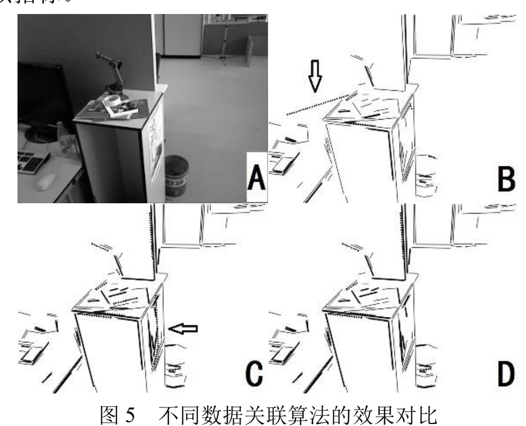


图 5　不同数据关联算法的效果对比

Fig.5　Comparison of effect of different data association algorithms

图 5 为实验运行时的系统截图，其中 A、B、C、D 分别为输入图像、算法 3、算法 2 和算法 1 得到的关联结果。关联结果图像中粗实线为关联对距离在一定阈值以内的三维线段的二维投影线段，视为正确关联；粗虚线为关联对距离超出阈值的二维投影线段，视为错误关联；细实线为图像中提取出的二维线段。图像 B、C 中的黑色箭头指明了重点分析区域。

图像 A 来自于数据集场景 2，由于算法 3 仅基于线段描述子 LBD，当地图中三维线段的优化结果存在误差时，算法 3 难以剔除这样的误差关联项，如图像 B 中箭头所指的二维投影线段为地图中误差较大的三维线段投影所得，算法 3 未能将其剔除。而算法 2 中仅根据投影线段的距离来检测关联对，当线段分布密集时会得到较多错误关联对，图像 C 中箭头指向处线段较为密集，此时算法 2 得到的错误关联对较多，而在本文提出的算法中则可以使用 Patch 确认来剔除这些错误关联。

表 1　线特征关联算法实验结果

Table 1　Experimental result of data association algorithms

数据集场景（1-4）		线特征关联算法		
		算法 1	算法 2	算法 3
1	错误关联帧比例	0.245%	1.09%	4.52%
	平均关联对数目	84	96	48
	平均算法时间（ms）	11.039	6.2899	12.316
	平均投影误差 G	1.360	2.083	1.736
2	错误关联帧比例	2.29%	12.94%	6.36%
	平均关联对数目	71	76	46
	平均算法时间（ms）	16.362	7.641	18.108
	平均投影误差 G	1.866	3.010	2.253
3	错误关联帧比例	1.95%	7.07%	9.24%
	平均关联对数目	126	126	97
	平均算法时间(ms）	25.514	10.094	18.148
	平均投影误差 G	2.159	2.908	2.749
4	错误关联帧比例	0.52%	2.17%	2.42%
	平均关联对数目	64	71	58
	平均算法时间（ms）	9.10	5.151	13.35
	平均投影误差 G	1.139	1.875	1.830

表 1 为 3.1 节中介绍的关联误差评价模块对各算法的输出结果记录，其中错误关联帧比例为 G、M 和 L 三项评价指标中任意一项超出阈值的帧占全部帧的比例，平均投影误差为评价指标 G 相对于每一帧的平均值。从错误关联帧比例和平均投影误差 G 来看，本文算法在四个数据集场景上的运行结果均为最低，显著提高了关联算法的正确率；从关联对数目来看，本文算法的关联对数目稍小于算法 2，但其中错误关联对数目最少，而过多的错误关联对会明显降低 SLAM 系统的鲁棒性；从算法时间上来看，本文算法在四个实验场景中每帧的运行时间均在 30ms 以内，其中二维线特征数目最多的场景 3 和包含闭环的场景 2 中的算法运行时间较长，场景 2 中相机的闭环运动使线特征数据关联算法的输入数据中增加了很多被遮挡的三维线特征，但由于采用了 2.1 节所描述的缩小候选关联集合的策略和 2.3 节中的迭代优化方法，当输入的二维线特征或三维线特征数量增多时，算法规模并不会急剧扩大，算法仍然能够保证实时运行。其他算法中算法 2 速度最快，但其错误关联帧比例在不同运行场景下变化较大，难以处理线段密集的场景；算法 3 是较为常用的基于线段的 LBD 描述子的关联方法，其错误关联帧比例在不同场景中较为稳定，但此算法耗时较长、得到的关联对数目最少，由之前的单帧运行情况来看，该算法不易处理输入数据中的误差项和纹理稀疏的线段。

为了分析不同线特征关联算法对 SLAM 系统性能的影响，本文在同一套 SLAM 系统中分别采用基于算法 1、算法 2 和算法 3 的局部地图跟踪模块进行了实验。由于算法 2 在场景 2 中的错误关联帧比例过高，仅对数据集场景 1、3、4 进行实验，SLAM 系统得到的相机位姿的相对误差见表 1。

表 2 不同线特征关联算法的相对位姿误差

Table 2 Relative pose error of different data association algorithms

数据集场景		线特征关联算法		
		算法 1	算法 2	算法 3
1	RMSE (m/s)	0.0304	0.0436	0.0484
	中值(m/s)	0.0252	0.0371	0.0413
	RMSE (deg/s)	0.5587	0.8092	0.8529
	中值(deg/s)	0.3599	0.3876	0.4413
3	RMSE (m/s)	0.1253	0.1689	0.1403
	中值(m/s)	0.1218	0.1676	0.1342
	RMSE (deg/s)	1.5433	3.7659	1.9632
	中值(deg/s)	1.1089	2.4806	1.3593
4	RMSE (m/s)	0.0158	0.0176	0.0267
	中值(m/s)	0.0112	0.0128	0.0118
	RMSE (deg/s)	1.3184	1.7336	1.160
	中值(deg/s)	0.4179	0.7874	0.4864

表 2 中的位姿轨迹相对误差表明本文提出的线特征关联算法提高了单目线特征 SLAM 系统的整体性能，相对于仅使用线特征几何关系的迭代最近邻关联算法和仅使用线段描述子的基于 LBD 的关联算法，本文提出的算法具有更高的稳定性和准确性。

4 结束语

线特征的数据关联是单目线特征 SLAM 面临的一个重要问题。近年来点线结合的单目 SLAM 系统因其准确性和鲁棒性的优势引起了较多学者的注意[10,15,20]，但这些与线特征相关的基础问题限制了线特征优势的发挥。本文针对难度较大的单目线特征 SLAM 系统，提出了一种新的线特征数据关联算法。本文提出的算法融合了最近邻匹配和基于线段端点图像 Patch 的错误关联剔除策略，使用迭代的处理方式。在公共数据集上的实验表明，本文提出的迭代线特征关联算法在关联准确率和关联对数目上均达到了较好的性能要求。后续工作中将进一步研究点特征和线特征的联合关联算法，研究提升点线融合 SLAM 性能的方法。

参考文献：

[1] Klein G, Murray D. Parallel tracking and mapping for small AR workspaces [C]// Proc of IEEE and ACM International Symposium on Mixed and Augmented Reality. Piscataway, NJ: IEEE Press, 2007: 225–234.

[2] Mur-Artal R, Montiel J M M, Tardós J D. ORB-SLAM: a versatile and accurate monocular SLAM system [J]. IEEE Trans on Robotics, 2015, 31 (5): 1147-1163.

[3] Perdices E, López L M, Cañas J M. LineSLAM: visual real time localization using lines and UKF [C]// Proc of the 1st Iberian Robotics Conference. 2014: 663–678.

[4] Zhang Guoxuan, Suh I H. SoF-SLAM: segments-on-floor-based monocular SLAM [C]// Proc of IEEE//RSJ International Conference on Intelligent Robots and Systems. Piscataway, NJ: IEEE Press, 2010: 2083-2088.

[5] Zhang Lilian, Koch R. Hand-held monocular SLAM based on line segments [C]// Proc of Machine Vision and Image Processing Conference. Piscataway, NJ: IEEE Press, 2011: 7-14.

[6] 武涛, 孙凤池, 苑晶, *et al.* 一种基于线段特征的室内环境主动 SLAM 方法 [J]. 机器人, 2009, 31 (2): 166-70. (Wu Tao, Sun Fengchi, Yuan Jin, *et al.* An active SLAM approach based on line segment feature in indoor environment [J]. Robot, 2009, 31 (2): 166-70.)

[7] Zhang Guoxuan, Suh I H. A vertical and floor line-based monocular SLAM system for corridor environments [J]. International Journal of Control Automation & Systems, 2012, 10 (3): 547-557.

[8] Zhou Huizhong, Zou Danping, Pei Ling, *et al.* StructSLAM: Visual SLAM With Building Structure Lines [J]. IEEE Trans on Vehicular Technology, 2015, 64 (4): 1364-1375.

[9] Smith P, Reid I, Davison A. Real-Time Monocular SLAM with Straight Lines [C]// Proc of British Machine Vision Conference. Berlin: Springer,



2006: 17-26.

[10] Pumarola A, Vakhitov A, Agudo A, *et al.* PL-SLAM: Real-time monocular visual SLAM with points and lines [C]// Proc of IEEE International Conference on Robotics and Automation. Piscataway, NJ: IEEE Press, 2017: 4503-4508.

[11] Wang Zhiheng, Wu Fuchao, Hu Zhanyi. MSLD: a robust descriptor for line matching [J]. Pattern Recognition, 2009, 42 (5): 941-53.

[12] Zhang Lilian, Koch R. An efficient and robust line segment matching approach based on LBD descriptor and pairwise geometric consistency [J]. Journal of Visual Communication & Image Representation, 2013, 24 (7): 794-805.

[13] Zhang Lilian, Koch R. Structure and motion from line correspondences: Representation, projection, initialization and sparse bundle adjustment [J]. Journal of Visual Communication & Image Representation, 2014, 25 (5): 904-915.

[14] Bartoli A, Sturm P. The 3D line motion matrix and alignment of line reconstructions [J]. International Journal of Computer Vision, 2004, 57 (3): 159-178.

[15] Gomez-Ojeda R, Briales J, Gonzalez-Jimenez J. PL-SVO: semi-direct monocular visual odometry by combining points and line segments [C]// Proc of IEEE//RSJ International Conference on Intelligent Robots and Systems. Piscataway, NJ: IEEE Press, 2016: 4211-4216.

[16] Kümmerle R, Grisetti G, Strasdat H, *et al.* G2o: a general framework for graph optimization [C]// Proc of IEEE International Conference on Robotics and Automation. Piscataway, NJ: IEEE Press, 2011: 3607-3613.

[17] Akinlar C, Topal C. EDLines: A real-time line segment detector with a false detection control [J]. Pattern Recognition Letters, 2011, 32 (13): 1633-42.

[18] Witt J, Weltin U. Robust stereo visual odometry using iterative closest multiple lines [C]// Proc of IEEE//RSJ International Conference on Intelligent Robots and Systems. Piscataway, NJ: IEEE Press, 2013: 4164-4171.

[19] Sturm J, Engelhard N, Endres F, *et al.* A benchmark for the evaluation of RGB-D SLAM systems [C]// Proc of IEEE//RSJ International Conference on Intelligent Robots and Systems. Piscataway, NJ: IEEE Press, 2012: 573-580.

[20] Yang Shichao, Scherer S. Direct monocular odometry using points and lines [C]// Proc of IEEE International Conference on Robotics and Automation. Piscataway, NJ: IEEE Press, 2017: 3871-3877.